DYNAMIC KEY GENERATION DURING A COMMUNICATION INSTANCE OVER GSM

Joseph Zalaket[1], Khalil Challita[2]

[1] Holy-Spirit University of Kaslik, Faculty of Engineering,
446, Jounieh, Lebanon
josephzalaket@usek.edu.lb

[2] Notre-Dame University, Louaize, Faculty of Natural & Applied Sciences,
72, Zouk Mikael, Zouk Mosbeh, Lebanon
kchallita@ndu.edu.lb

Abstract

Mobile phone may become the protagonist of the new electronic technology. If we compare it with that of other technologies, the infiltration rate of mobile phones in the world is extremely high, both in cities than rural communities of the most of the countries. According to estimates made by the International Telecommunication Union the access to mobile networks is growing much faster than the access to Internet. This emergence has led many companies to allow new activities which were previously running strictly over the Internet to run over the mobile network such as the electronic payment. These circumstances make the security of mobile communication a priority to preserve the authentication, confidentiality and integrity of data sent between subscribers and mobile network. In this paper, we propose a dynamic key generation for the A5 GSM encryption algorithm to enforce the security and protect the transferred data. Our algorithm can be implemented over any GSM generation GSM/3G/4G.

Key words: mobile communication, encryption, GSM, A5 algorithm

1 Introduction

The use of mobile phones became vital in our everyday life. Most of the mobile phones are running through GSM (Global System for Mobile Communication). The security of the GSM is based on three main algorithms [1], [9]. A3 is an authentication algorithm, A8 is a key generation algorithm, these two algorithms are based on a hash function that takes as input a 128-bit key Ki stored in SIM card and at the network side and a random number ‘RAND’ of 128-bit to generate a 32-bit RESponse that will either verify the identity of the subscriber or deny it. These same inputs are given to A8 to generate a key of 64 bits that will be used to encrypt all the messages using an A5 algorithm. There are several versions of A5: A5/1, A5/2 and A5/3. The one implemented in the second generation and providing an acceptable level of security is A5/1.

Although there are several cryptographic algorithms used to provide security in GSM, this system is not completely secure since both A3/A8 algorithm and the A5/1 [10], A5/2 [14] and A5/3 [12] algorithms were broken, and the key used in ciphering was revealed sometimes in few seconds.

Trying to avoid attacks against different A5 versions we propose the dynamic generation for temporary keys that can be changed for multiple times during one communication instance. The dynamic key generation will replace the static single shared key originally stored on SIM card. Our method will require changing the architecture of SIM cards which can be a lack for its application as many users are running old cards and it is unreasoned to ask them to simply change their cards. To treat this lack we propose an algorithm that supports our extended SIM card architecture as well as the existing one.

2 Background

Works are done for making the authentication mutual from both the network and the mobile subscriber (MS) in such a way to avoid these attacks and many others [10-12].

The authors in [2] used the public key encryption to introduce a new mutual authentication method between the MS and the Visitor Resource Locator (VLR): Proxy signature.

In this method, the Home Resource Locator (HLR) will delegate the Mobile Station (MS, subscriber) the power to sign the nonce (random number) generated by the VLR and the VLR can verify the signature based on the HLR public key knowing that the MS is the one who signed. This model provides user privacy and non-repudiation features. Key management is easy since only one key (Public key of HLR) must be managed.

Authentication phase is divided into two parts: on-line authentication and off-line authentication. In the on-line authentication phase, the process requires that VLR must connect to HLR whenever MS demands authentication. However, without connecting to HLR to save authentication time and provide fault tolerance, off-line authentication is performed by VLR locally according to the parameters obtained from HLR in advance. Note that the first authentication request must be performed online and the subsequent authentication requests can be continually performed off-line.

A new protocol published by the IETF, “Extensible Authentication Protocol” or EAP-SIM, here knowing the weakness of the key used in encryption (64-bit Kc), they tried to make use of the 5 triplets allowed for the GSM network to send (RAND, SRES, Kc) to generate several RAND numbers and therefore produce more than one session key. All these keys will be then combined in one master key that is stronger which will be then used as input to the encryption algorithm [3].

As per [4], their proposed security system provides an authenticated session key distribution protocol between the authentication center AuC and the mobile station MS for every call attempt made by a MS. At the end of an authenticated session key distribution protocol the identities are mutually verified between the AUC of a Public Land Mobile Network PLMN and the Subscriber Identity Module SIM of a MS as well as the session key for call encryption is distributed to the MS.

A self-concealing mechanism is introduced in [5], the purpose here is to reduce the entries of the GSM databases (VLR and HLR) by discarding the bulky database and create valuable improvements for portable communication systems. In this approach, the shared secret is concealed by the authentication server and only the authentication server has the private key to open the shared secret.

The new concept initiates several positive changes. First, the sensitive and large database can be discarded. Consequently, this prevents hacker attacks to the database and reduces maintenance demand for the server.

A warrant is used to guarantee the user's access rights; an issue is not addressed in the conventional challenge-response scheme.

When it comes to our protocol, we profited from new variables that already exist in the GSM and used but not in the security algorithms such as TMSI, LAC and a new random number.

3 Dynamic key generation

Note that, any change arising in the algorithms currently implemented on the SIM card requires the generation of new SIM cards for all users and that could be inconvenient for many of them. That's why the protocol we are proposing will support two versions of SIM cards, the old one that is currently used and therefore, the subscribers that don't want to change their SIMs will keep benefiting from the security features already provided by the GSM while the holders of the new generated SIMs will profit from better security described in the following sections.

3.1 Key generation process

GSM network sends a 128-bit RAND number to the MS, they both apply the A3 (authentication algorithm) that will accept this RAND and a 128-bit key Ki stored in the SIM and known to the GSM as input to generate a 32-bit RESponse number. The MS sends this RES to the GSM network that will compare it to the response number it generated, if they match, the user is authenticated and they can now apply the A8 algorithm that takes Ki and RAND as input and generates the 64-bit key Kc that will be used in the ciphering algorithm A5.

Now that the Kc is generated, new core elements take part in the new protocol: 32-bit TMSI, 64-bit IMSI, 16-bit LAC and a new number NEWRAND that is randomly generated by the GSM.

A GSM conversation is sent as a sequence of frames every 4.6 millisecond [6] and therefore we chose to generate the NEWRAND every 10 x 4.6 ms that means every 46 milliseconds.

Once this new random number is generated, the timestamp (dd/mm/hh/mm/ss) will be divided by the NEWRAND.

This time variable has a 24-hour format to avoid having the same timestamp twice a day. A random interval was chosen to decide whether or not to use this NEWRAND. If the division result was within the interval then the NEWRAND is sent for use by both the MS and the GSM otherwise, the old value of NEWRAND is used. This interval can be left for the operator to decide its boundaries for better security and to avoid having any pattern. But in our simulation the range of values was chosen to be within [20790, 977890].

Once the NEWRAND is generated and sent, the main function called: KeyGen() will use it with other parameters to generate an enhanced security key.

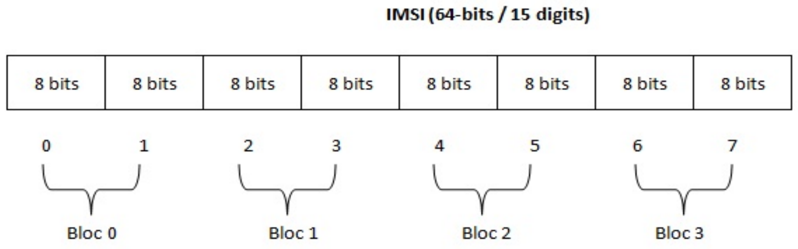


Figure 1. IMSI blocks

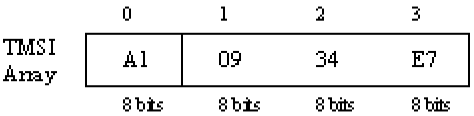
The main function KeyGen() takes seven parameters: TMSI, LAC, IMSI, NEWRAND, Kc, version, NewKeyGenerated.

- TMSI is the 32-bit or 8 digits value that is changing on VLR change or if the interval of time set for it expires.
- 16-bit LAC or 4 digits, location area code that will be updated at each location change. A Location Update Procedure is triggered.
- IMSI is 15 digits number (64-bits); it is the main identifier for the MS. In our protocol we will divide it (logically) into 4 blocks and based on the

NEWRAND we will select which block can be used to generate the new key as shown in Figure 1.

- NEWRAND already generated as described in the previous subsection
- Kc is the original 64-bit key generated using the A8 algorithm.
- Version is set to 1 in this protocol because it is the new version of SIM card, for the previous cards it will be set to 0. This function will only run if the version was 1.
- NewKeyGenerated is the new 64-bit key generated at each execution of the function KeyGen() and that will be used as input to the A5 ciphering algorithm.

All the values used as identifiers such as TMSI, LAC, IMSI and the key are originally sent in hexadecimal format [7]. So each one of them is converted to binary format and stored in a vector of bytes.Example: TMSI is of 8 hexadecimal digits: A1 09 34 E7, it will be converted to binary and stored in a 4 bytes vector as in Figure 2:



	0	1	2	3
TMSI Array	A1	09	34	E7
	8 bits	8 bits	8 bits	8 bits

Figure 2. TMSI array

So, in memory A1 will be stored as (10100001). Same applies for all these variables in term of conversion from hexadecimal to binary.

As we can see, TMSI is an array consisting of 4 cells. The possibilities of arranging these cells in different order are 4! = 24. For this purpose, we have created a static matrix that consists of 24 rows and 4 columns. In each column is stored an index from 0 to 3. This matrix plays the role of an index to the TMSI array in order to accelerate the generation.

The remainder of the division of NEWRAND by 23 will determine the index of the row of the matrix to be chosen. Let's say the NEWRAND modulo 23 gave 5 as a result. Looking at the row of index 5, the columns contain respectively 0,3,1,2. Therefore the rearranged TMSI array will have as elements: TMSI[0], TMSI[3], TMSI[1] and TMSI[2] respectively. And thus the rearranged TMSI will be A1 E7 09 34.

The remainder of the division of NEWRAND by 23 will determine the index of the row of the matrix to be chosen. Let's say the NEWRAND modulo 23 gave 5 as a result. Looking at the row of index 5, the columns contain respectively 0,3,1,2. Therefore the rearranged TMSI array will have as elements: TMSI[0], TMSI[3], TMSI[1] and TMSI[2] respectively. And thus the rearranged TMSI will be A1 E7 09 34.

The LAC array is of 16 bits (2 hexadecimal digits), for example: F2 56.

The IMSI is divided into 4 blocks of 16 bits. Also based on this NEWRAND we will decide which block to use. The remainder of NEWRAND divided by 4 will determine the number of the block. If NEWRAND modulo 4 gave 2 as remainder, the block of index 2 of IMSI is chosen which means the IMSI[4] and IMSI[5] are used.

A new array of 4 bytes is introduced, which is the combination of the LAC array and the two chosen cells of the IMSI. This 32-bit array is XORed bit by bit with the rearranged TMSI array.

Since NEWRAND is even, the New Output will be swapped with block 1 of Kc and therefore the new key generated consists of the block 0 of Kc concatenated with the new output (32-bits) as described in Figure 3.

This whole process is repeated:

- Every 46 milliseconds
- Whenever the TMSI value is updated
- Once a Location Update Procedure is taking place

So we have multiple keys that will be generated every while. Each time a new key is generated, the A5 will call its main function using the newly generated key as input.

3.2 New algorithm for supporting existent SIM cards

Now that the multiple keys generation is solved, we still have to make the new protocol compatible with both versions of SIMs. This must be taken into consideration from the GSM network view given that it would be the one responsible of generating the NEWRAND and sending it to the MS. Since the implementation of the global function grouping all three algorithms (A3, A8, A5) is not made clear by GSM community, the following is a theory of how the protocol should be put into practice

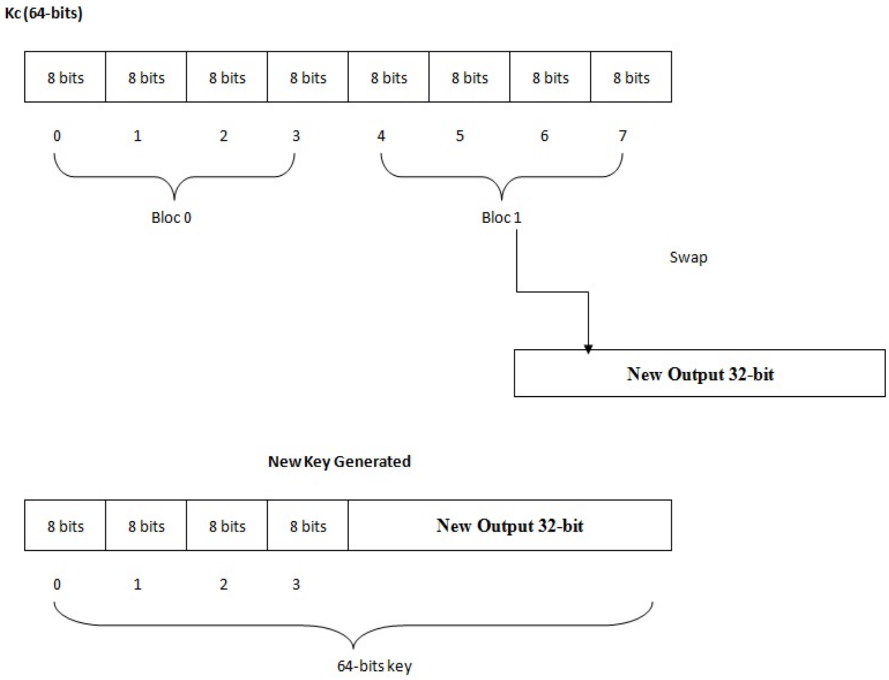


Figure 3. New Key generation

We assume that the major function that combine all the elements is a function called F() having several parameters (key, TMSI, LAC, IMSI). So if the values of the last three parameters are not sent then we know that we are using the old version of SIM cards and therefore the usual protocols will be followed. If these values are sent as parameter arguments then we are using a new version of SIM cards and therefore the KeyGen() function early described will be applied.

The below pseudo code explains the idea and how default value of null will be assigned to non sent parameters in order to be compatible with the current SIMs:

```
F(char Key,char TMSI=Null,char LAC=Null,char IM-
SI=Null)

{       If no values are set for TMSI, LAC and IMSI

        Use the old protocol and algorithms

        e.g. call g (key)

        Else
Use A3/A8 and then call the new function KeyGen

e.g. call h (Key, TMSI, LAC, IMSI)

}
```

3.3 New SIM card architecture

Further to the functions that are newly embedded in the system, we need to take into consideration the elements that were added and were not available in the previous version. The NEWRAND for example is used in our new protocol and therefore we need to store it in the SIM. For this purpose, the new SIMs will hold a new register for the NEWRAND value; each new value will override the previous one.

The LAI value is updated using a BCCH (broadcast control channel) with ongoing conversation, for this reason we will be using this same channel to send the NEWRAND whenever it is generated and the decision was to send it.

The function that is responsible of the NEWRAND generation and sending will be implemented on the network side only, while the function executed for the key generation should be available on both SIM and network.

4 Implementation

A version of our algorithm is implemented using the C language. The simulation has been done on a PC having the following specifications:

Intel Core i3 CPU, 2.40 GHz - 3 GB of DDR2 RAM - Windows 7 32-bit OS - 500 GB Hard Disk

Note that the capacity of the used machine is not important as the algorithm uses trivial system resources such that: only 30 bytes of RAM in addition to which is used by the A5/1 algorithm, 2 bytes of physical memory (register to store the random number on the SIM card side) and about 0.001ms of CPU time to generate a new key. Even though, a new implementation of this prototype is currently prepared to run on mobile phones using Android OS.

A simplified simulation of an ongoing call running during less than 0.5 second gives the following observations:

- Eight new keys have been generated during this short and the same key has been sent twice during two different sent sessions.
- The same timestamp has been assigned for all these tasks as they are done during less than one second (the necessary time to get a new timestamp).

5 Discussion

Using a single key during a whole mobile communication or even for a long period of time made it easy for cryptanalyzers to reveal the key and get all the information shared between the MS and the network. Whereas in our protocol we have more than one key that is being used within the same conversation and its way of generation is not patterned. The location update depends on whether the user is moving or not, the TMSI change depends also on the network implementation in addition the random number periodically generated is not always used so an eavesdropper might have to wait a long period of time to get the random used.

The timestamp varies every second, and its XOR Result with the NEWRAND cannot be detected since it is done on the network side. Also the range in which this result is tested can be set by the operators making it trickier for attackers. The generation of the number is done randomly so guessing this number is practically very difficult to achieve.

As known, weakness in GSM was that algorithms were never published, and therefore their security level was not evaluated. Even if our protocol is published, the key element that is changing is the NEWRAND number. So an attacker has to detect the key that was originally generated and then to track every random number that is being sent irregularly. In case he misses one of the random numbers, he won’t be able to decrypt the message.

The NEWRAND number is generated every 46 ms but the time execution of the algorithms is still the same because they are not modified. So in case we decided to generate this random less frequently (make time more than 46 ms), catching it becomes harder while generating this number in less than 46ms will make it simpler.

This new feature we offer is not mandatory; we are not forcing the users to change their SIM cards. They can keep the old ones and benefit from the same security features already offered by GSM. For those who are seeking a better level of security will choose the new version of SIM cards which can also be offered by default for new subscribers.

6 Conclusion

The GSM community pretends that its used algorithms are safe and secure. By contrast, many successful attacks have been done on each of these algorithms and this is due in the first place to the fact that the implementation of these algorithms were kept secret and they were reversed engineered.

Several attempts were made to enhance security algorithms and several propositions were made for both authentication and encryption algorithms.

We, in our turn, have proposed a new protocol for generating multiple keys in the same conversation, so each stream might be encrypted using a different session key than the previous one. The strength of our protocol lies in the fact that we depend on several variables that are very hard to be detected or guessed by an eavesdropper, the encrypted TMSI, the LAC changing at each location change and a random generated by the network but that is not constantly sent to avoid any pattern. Also the new protocol supports two versions of SIM cards. The subscribers holding the old versions will profit from the usual security features while those using the new version will benefit from a higher level of security.

We have simulated these functions on a normal PC using the C programming language, the same used in the other algorithms implementation.

Our strategy was not to hide the proposed algorithm from cryptanalyzers, but rather to use a combination of dynamic parameters that render the task of attackers semi-impossible. (e.g. the technique of public key or asymmetric key cryptography)

The A5/1 and A5/2 algorithms were breached, and the key (Kc) used in ciphering was revealed. To defend, GSM engineers implemented a 3rd generation system, but also kept secret. They claim enhancing the security authentication with their new Authentication and Key Agreement protocol (AKA) also with the new block cipher algorithm A5/3 [8] based on KASUMI which has been also attacked [12], [13].

References

1. Brakan E., Biham E. and Keller N., 2003, *Instant Ciphertext-only cryptanalysis of GSM encrypted communication*, CRYPTO 2003: 600-616
2. Haverinen H., Nokia Ed. and Salowey J., Ed. Cisco Systems, 2006, *RFC 4186 "Extensible Authentication Protocol Method for Global System for Mobile Communications (GSM) Subscriber Identity Modules (EAP-SIM)"*, IETF
3. Duraiappan C., Zheng Y., *Enhancing Security in GSM*, University of Wollongong
4. Wei-Bin Lee and Chang-Kuo Yeh, 2008, *A Self-Concealing Mechanism for Authentication of Portable Communication Systems*, International Journal of Network Security, Vol.6, No.3, PP.285–290
5. Antipolis S., 2003, *Identity protection using P-TMSI for GPP/WLAN interworking*, 3GPP,TSG,SA WG3 Security – S3#26
6. Biryukov A., Shamir A., Wagner D., 1999, *Real Time Cryptanalysis of A5/1 on a PC*
7. http://www.gsm-security.net/papers/a51.shtml
8. GSM 2000 Joint GSMA TSG SA WG3 Working party, *Requirements Specification for the GSM A5/3 Encryption Algorithm, version 0.5*

9. Barkan E., Biham E., Keller N., 2008, *Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication,* J. Cryptology 21(3): 392-429
10. Barkan E., Biham E., 2002, *Conditional Estimators: An Effective Attack on A5/1. Selected Areas in Cryptography*: 1-19
11. Ekdahl P., Johansson T., 2003, *Another attack on A5/1. IEEE Transactions on Information Theory* 49(1): 284-289
12. Dunkelman O., Keller N., Shamir A., 2010, *A Practical-Time Attack on the A5/3 Cryptosystem Used in Third Generation GSM Telephony.* IACR Cryptology ePrint Archive 2010
13. Dunkelman O., Keller N., Shamir A., 2010, *A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony*. CRYPTO 2010: 393-410
14. Paglieri N., Benjamin O., 2011, *Implementation and performance analysis of Barkan, Biham and Keller's attack on A5/2,* Ensimag- Grenoble Institute of Technology - INP